INFO 301 – DEV WEB – JAVASCRIPT – TP 3

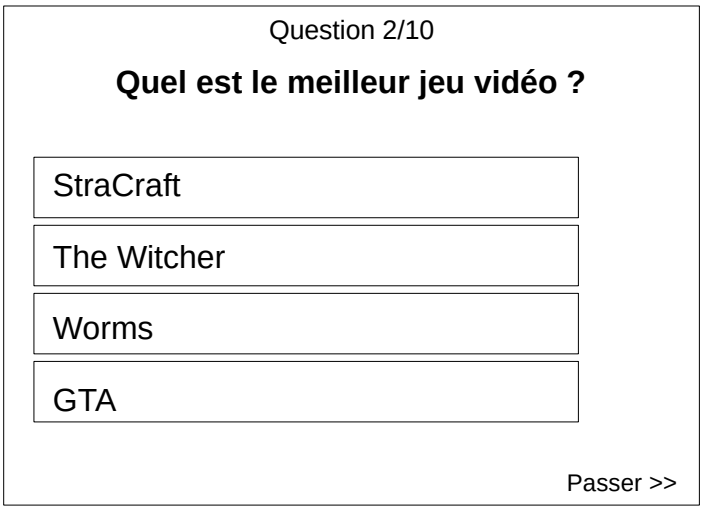
Objectif du TP :

Réaliser un QCM à l'aide de VueJS, version simplifiée.

- Les questions sont tirées aléatoirement.
- Dès que la réponse est choisie (clic), la question suivante est affichée.
- Il n'y a pas de retour en arrière possible.
- Les questions peuvent être passées.
- Une fois terminé, le nombre de bonnes réponses doit être affiché.

Pour aller plus loin :

- L'application commence par une page de connexion où le nom de l'utilisateur est demandé.
- Récupérez (en AJAX) les questions sur le serveur (une par une, ou toutes en une seule fois).
- Enregistrez les réponses de l'utilisateur sur le serveur.



Utilisation de VueJS :

Pour simplifier, VueJS sera utilisé en mode simplifié : il suffit d'importer le fichier « vue.global.js ». Cela ne permet pas d'utiliser les composants. L'objectif est ici de profiter du principe de **réactivité**.

Référez vous au code du CM3.

La doc est bien faite : https://vuejs.org/

Réalisez votre application de manière progressive.

```
<body>

  <main id="app" >
      {{ x }}
  </main>

  <script src="axios.min.js"></script>
  <script src="vue.global.js"></script>
  <script src="main.js" defer></script>

</body>
```

```
const { createApp } = Vue

createApp({
  data() {
    return {
      x: 42
    }
  },
  methods: {
    inc() {
      this.x++
    }
  },
  computed: {
  },
  mounted() {
    // code éxécuté au chargement
  }
}).mount('#app')
```

Fonctionnalités utiles :

Affichez le contenu d'une variable :
`{{ x }}`

Créez plusieurs tags avec une boucle :
`<li v-for="q in questions">{{ q }}</li>`

Réagir à un événement (exécution de la fonction pass)
`<button @click="pass">Passer</button>`

Afficher ou cacher un élément en fonction d'un booléan
`<div v-if="show"></div>`

Connecter un champ à une variable
`<input v-model="lastname" />`